Resilient model predictive control framework for networked industrial cyber-physical systems with multi-disturbance adaptation

Journal of Intelligent Manufacturing and Special Equipment

Received 24 December 2025
Revised 1 February 2026
Accepted 1 March 2026

Milad Rahmati
Independent Researcher, Los Angeles, California, USA, and
Nima Rahmati
Independent Researcher, Yazd, Iran

Abstract

Purpose – The purpose of this study is to develop a resilient model predictive control (RMPC) framework capable of maintaining optimal control performance in networked industrial cyber-physical systems subject to unpredictable disturbances, communication delays and actuator faults. The proposed approach aims to enhance the resilience, stability and reliability of intelligent manufacturing processes by integrating adaptive fault handling, disturbance compensation and real-time constraint adjustment within a unified predictive control architecture.

Design/methodology/approach – The proposed RMPC framework employs a disturbance observer-based adaptive MPC structure enhanced by a resilience index for real-time constraint regulation. It integrates multi-disturbance estimation, adaptive fault reconfiguration and stability-guaranteed optimization using Lyapunov-based analysis. Simulation studies on a representative industrial heating, ventilation and air conditioning (HVAC) control benchmark validate the method's capability to sustain control performance under faults and network delays, demonstrating the framework's computational efficiency and applicability to intelligent manufacturing systems.

Findings – Results demonstrate that the proposed RMPC significantly enhances fault tolerance, stability and tracking performance compared with conventional MPC and proportional-integral-derivative controllers. The framework reduces cumulative tracking error by up to 65%, achieves 50–60% faster recovery from actuator faults, and maintains constraint satisfaction under multiple disturbances. These findings confirm the RMPC's suitability for complex, networked manufacturing environments requiring robust, adaptive and computationally efficient control solutions.

Originality/value – This study introduces a unified resilient predictive control architecture that integrates disturbance estimation, adaptive constraint regulation and fault recovery into a single optimization framework. Unlike traditional MPC approaches, the proposed method explicitly quantifies resilience and guarantees stability under network-induced uncertainties. The originality lies in bridging resilient control theory with intelligent industrial applications, providing a practical pathway for deploying fault-tolerant, adaptive controllers in next-generation smart manufacturing and cyber-physical systems.

Keywords Robust control, Resilience, Model predictive control, Disturbance rejection, Cyber-physical systems, Networked control systems

Paper type Research article

1. Introduction

1.1 Background and motivation

Industrial automation has undergone a substantial transformation with the advent of cyber-physical systems (CPS), which combine computational intelligence with physical processes

Funding: This research did not receive any specific grant from funding agencies in the public, commercial or not-for-profit sectors.

Competing interests: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

Journal of Intelligent Manufacturing and Special Equipment
Vol. 7 No. 2, 2026
pp. 153-174
Emerald Publishing Limited
e-ISSN: 2633-660X
p-ISSN: 2633-6596
DOI 10.1108/JIMSE-12-2025-0036

through the tight integration of sensors, actuators, embedded controllers and communication networks. These systems are foundational to modern infrastructures such as smart grids, intelligent transportation systems, smart manufacturing and building energy management. In such settings, control systems must maintain high levels of performance while operating in environments characterized by uncertainty, network-induced disturbances, hardware failures and varying load profiles.

Model predictive control (MPC) has emerged as a powerful paradigm for managing such complex, multivariable control problems. Unlike classical proportional-integral-derivative (PID) controllers or linear quadratic regulators, MPC explicitly predicts future system behavior and solves an online optimization problem to determine the optimal control input at each sampling instance. This ability to anticipate system evolution makes MPC well-suited for CPS environments that demand high adaptability and precision.

However, MPC's deployment in real-world CPS is fraught with practical challenges. Industrial control systems often operate over networked infrastructures that introduce latency, packet loss and jitter – phenomena that degrade control performance and may destabilize the system if not properly addressed. Furthermore, these systems are subject to external disturbances and internal faults that can manifest unpredictably, necessitating real-time robustness and resilience.

1.2 Problem statement

While conventional MPC techniques assume ideal operating conditions and deterministic models, industrial CPS are inherently nonideal. Controllers must operate under limited communication bandwidth, unpredictable time delays, sensor/actuator faults and heterogeneous disturbance sources. These include step load changes, actuator saturation, environmental noise and cybersecurity threats that can compromise sensor integrity or disrupt control signals. In such cases, standard MPC may yield suboptimal or even unsafe behavior due to its inability to adapt dynamically to unforeseen disturbances or faults.

Despite substantial research in robust and stochastic MPC variants, there remains a theory-practice gap. Many existing solutions require accurate disturbance models, are computationally intensive, or lack real-time feasibility. Moreover, few frameworks incorporate *resilience* as a design objective – defined here as the system's ability to recover performance following a disturbance or fault without requiring full system reinitialization or human intervention. This gap presents a critical bottleneck for industrial deployment, particularly in settings where safety, continuity, and autonomous operation are paramount.

1.3 Significance and national relevance

Resilient control architectures are vital to the advancement of industrial autonomy and smart infrastructure. In energy-critical applications such as load frequency control in microgrids, thermal regulation in HVAC systems or torque control in smart factories, the inability to cope with system-level disturbances can lead to cascading failures or operational shutdowns. Furthermore, such resilience aligns with broader societal goals, including energy efficiency, carbon footprint reduction and fault-tolerant infrastructure – areas of national priority in many developed countries.

With increasing digitization, cyber-physical vulnerabilities in industrial control systems have attracted significant attention. Attacks exploiting timing channels or injecting false data into control loops pose nontrivial risks. Therefore, the convergence of control resilience, network awareness, and computational efficiency is no longer optional – it is a requirement. Embedding resilience at the control layer – without relying solely on external failover mechanisms – provides an avenue for improving systemic reliability, availability and performance continuity.

1.4 Objective and scope

This research proposes a resilient model predictive control (RMPC) framework specifically tailored for networked industrial CPS exposed to multi-source disturbances. The core contributions of this study are.

(1) Development of a discrete-time linear time-invariant (LTI) system model for networked CPS with augmented disturbance and fault variables.

(2) Integration of a disturbance observer to estimate and compensate for real-time perturbations without prior disturbance modeling.

(3) Implementation of a fault-tolerant control (FTC) scheme that adapts the prediction horizon and input constraints in response to detected anomalies in sensor/actuator data.

(4) Real-time validation of the proposed RMPC framework through a case study on HVAC load control, showcasing its effectiveness under network latency, fault injection and disturbance rejection benchmarks.

The proposed approach provides both theoretical innovation and practical viability, bridging the performance and safety gaps in deploying advanced control algorithms in complex industrial environments.

1.5 Contributions

This article makes the following specific contributions to the field of control systems engineering.

(1) It introduces a resilience-centered design paradigm within the MPC framework, focusing on performance retention and recovery under nonideal networked control conditions.

(2) It presents a multi-layer control architecture that combines predictive optimization with a lightweight fault-detection and disturbance estimation mechanism, compatible with real-time implementation constraints.

(3) It formulates an adaptive MPC optimization problem that dynamically reconfigures its constraints and cost functions based on system condition feedback.

(4) It offers a comprehensive performance evaluation using industry-standard performance metrics (e.g. settling time, steady-state error, overshoot, control energy) across varying operational scenarios, contrasting the RMPC with traditional MPC and PID methods.

(5) It lays a foundation for extending the RMPC architecture to distributed CPS with heterogeneous communication topologies, addressing scalability in future industrial settings.

By addressing resilience at the control level and incorporating real-time fault adaptation strategies, this article provides a robust solution for industrial CPS and serves as a stepping stone toward safer and more autonomous industrial operations.

2. Related work

2.1 Model predictive control in industrial applications

MPC has gained widespread adoption in process industries, aerospace, power systems and automotive engineering due to its explicit handling of multivariable systems and constraints. Classic MPC methods, such as finite horizon quadratic programming, have shown promising results in regulation and tracking control (Maciejowski, 2002). Early applications focused on chemical process control and advanced manufacturing, where constraints on temperature, pressure and chemical composition had to be respected in real time (Mayne *et al.*, 2000).

The integration of MPC into industrial CPS represent a growing trend in recent years. Several studies have shown the applicability of MPC in distributed energy systems (Ma *et al.*, 2020), smart buildings (Privara *et al.*, 2013) and automated transport systems (Swaroop *et al.*, 1994). However, these implementations often assume stable communication channels, accurate models, and negligible external disturbances. While these assumptions may hold in controlled environments, they fall short in real-world scenarios characterized by high variability and vulnerability to faults.

2.2 Robust MPC and fault-tolerant design

To enhance reliability, researchers have proposed robust MPC (RMPC) techniques that account for bounded uncertainty within the system model. These include min-max optimization (Kerrigan and Maciejowski, 2003), tube-based MPC (Raković and Mayne, 2005), and stochastic MPC (Gupta *et al.*, 2013), which explicitly consider system noise or probabilistic disturbances. While robust MPC provides guarantees on performance under uncertainty, it often introduces computational burdens due to the complexity of the worst-case optimization problem.

FTC has emerged in parallel as a mechanism to maintain functionality in the presence of faults, whether due to sensor degradation, actuator failures, or communication losses. Passive FTC relies on controller robustness alone, while active FTC involves fault detection and reconfiguration. Several works, such as (Zhang and Jiang, 2008) and (Noura *et al.*, 2009), have integrated fault detection modules with predictive control to mitigate failures. However, many of these approaches assume full observability of fault signatures or rely on redundant physical components, which may not be feasible in constrained industrial settings.

In a notable study, Amin *et al.* (2019) reviewed anti-surge control systems in compressors and advocated for fault-tolerant integration. While their focus was limited to a single application domain, their findings highlight the importance of expanding FTC techniques to broader industrial CPS.

2.3 Networked control systems and time-delay compensation

The emergence of networked control systems (NCS) introduces new challenges such as packet loss, variable transmission delays and asynchronous updates. Time-delay and dropout-aware MPC methods have been developed to address these issues (Walsh *et al.*, 2002) (Zhang *et al.*, 2001). For example, delay compensation via state prediction has been explored in event-triggered MPC (Lehmann and Lunze, 2011). However, these strategies are sensitive to network topology and performance metrics and often require precise delay modeling, which may not be viable in dynamic industrial networks such as 5G-based or wireless mesh topologies.

Tipsuwan and Chow (2003) highlighted early issues in applying classical control to NCS, and many of their concerns – such as the effect of shared networks and jitter – remain open problems today. More recent work by Zhang *et al.* (2022) reviewed optimal control strategies in cyber-resilient power systems, but stopped short of proposing unified control models that integrate delay mitigation, fault tolerance and disturbance rejection.

2.4 Disturbance rejection and observer design

A crucial component in resilient control is the ability to detect and reject disturbances in real time. Traditional disturbance rejection relies on feedback error correction and signal filtering, as in PID controllers. More advanced designs incorporate disturbance observers such as Luenberger observers, extended Kalman filters or sliding mode observers (Franklin *et al.*, 1998) (Khalil, 2002).

For instance, Guo and Cao (2014) proposed an anti-disturbance control architecture for systems subject to multiple disturbances, showing improved stability in uncertain environments. Nevertheless, these methods are not seamlessly integrated into predictive control architectures and may suffer from estimator bias under abrupt fault conditions.

The concept of combining disturbance observers with predictive control is still underdeveloped. Existing work either focuses on observer-based state estimation for control purposes or treats disturbance compensation as a secondary feature. There remains a gap in combining these elements into a coherent, real-time adaptive control loop suitable for deployment in networked industrial environments.

2.5 Adaptive and learning-based control

Recent advances in machine learning and artificial intelligence have introduced adaptive control methods that learn system dynamics online. Reinforcement learning (RL) has been used to train policies that optimize control actions under uncertainty (Liu *et al.*, 2021). However, most RL algorithms are data-hungry, require extensive training and lack safety guarantees. While learning-enhanced MPC (L-MPC) frameworks are being developed (O'Donoghue *et al.*, 2016), they are rarely deployed in critical infrastructure due to interpretability and verification challenges.

Evolutionary algorithms and optimization-based strategies such as genetic algorithms and particle swarm optimization have also been applied to MPC tuning (Babuška and Verbruggen, 2003) (Deb *et al.*, 2002), but these methods are generally too slow for real-time applications in CPS with fast dynamics.

2.6 Resilient control and system recovery

The concept of resilience – distinct from robustness – is increasingly being recognized in control literature. While robustness refers to the ability to resist perturbation, resilience emphasizes the ability to recover from disruptions. In this context, McFarlane *et al.* (2013) explored product intelligence in industrial control, suggesting the embedding of resilience into physical and computational layers.

Despite this shift in understanding, formal resilience metrics for control systems are underdeveloped. There is no consensus on how to design controllers that dynamically adjust control laws, constraints or optimization horizons post-disturbance. Ivanov *et al.* (2019) surveyed control theory applications in supply chains and noted the lack of methods for ensuring resilience beyond static robustness, particularly under sequential disruptions.

Most relevant to this work, Wang *et al.* (2023) emphasized the "theory–practice gap" in transient control performance across industrial and military applications. They called for architectures that unify real-time responsiveness, disturbance rejection and post-fault performance stabilization.

3. Methodology

3.1 System description and modeling

We consider a discrete-time, LTI system that represents a typical industrial cyber-physical plant governed by control over a communication network. The state-space dynamics are affected by disturbances, delays and possible faults. The nominal system without disturbances is given by:

$$x(k+1) = Ax(k) + Bu(k) \tag{1}$$

$$y(k) = Cx(k) \tag{2}$$

where.

(1) $x(k) \in R^n$ is the system state vector,

(2) $u(k) \in R^m$ is the control input,

(3) $y(k) \in R^p$ is the output vector,

(4) A, B, and C are the system matrices with compatible dimensions.

To account for real-world uncertainties, we augment the model with.

(1) bounded additive disturbances $w(k) \in R^n$,

(2) sensor noise $v(k) \in R^p$,

(3) actuator faults $f_u(k)$,

(4) and delayed measurements due to the communication channel.

The augmented system is:

$$x(k+1) = Ax(k) + B(u(k) + f_u(k)) + w(k) \tag{3}$$

$$y(k) = Cx(k) + v(k)$$

To encapsulate network delay $\tau \in Z_{\geq 0}$, we introduce a delay operator Δ_τ such that:

$$u(k) = \Delta_\tau[u](k) = u(k-\tau) \tag{4}$$

The plant now becomes an NCS, and the controller must operate based on delayed and potentially corrupted state measurements.

3.2 Model predictive control formulation

MPC operates by solving a finite-horizon constrained optimization problem at each sampling instance k. The controller optimizes the predicted trajectory over a horizon N, but only applies the first control

$$\min_{\{u(k)\}} \sum_{i=0}^{N-1} \left(\|x(k+i|k)\|_Q^2 + \|u(k+i|k)\|_R^2 \right) + \|x(k+N|k)\|_P^2 \tag{5}$$

Subject to:

$$x(k+i+1|k) = Ax(k+i|k) + Bu(k+i|k), i = 0, \ldots, N-1 \tag{6}$$

$$x(k|k) = \widehat{x}(k) \tag{7}$$

$$x(k+i|k) \in \mathcal{X}, u(k+i|k) \in \mathcal{U} \tag{8}$$

Where.

(1) $Q \geq 0$ and $R \succ 0$ are weighting matrices,

(2) $P \geq 0$ is the terminal cost matrix,

(3) $\mathcal{X}$ and $\mathcal{U}$ define the constraint sets for state and input.

3.3 Resilience-oriented adaptation

To address resilience, we introduce adaptive constraint tightening and cost re-weighting mechanisms that depend on system status indicators (disturbance level, fault presence, network delay). Define a resilience index $\rho(k) \in [0, 1]$ representing confidence in system observability and controllability at time k. This is computed as:

$$P^{(k)} = \lambda_1 \cdot \frac{1}{1 + ||W(k)||2} + \lambda_2 \cdot \rceil^{-\delta\tau(k)} + \lambda_3 \cdot \eta(f_u(k)) \tag{9}$$

where $\lambda_1, \lambda_2, \lambda_3$ are tunable weights, and $\eta(\cdot)$ is a binary function indicating the presence of faults. The weights λ_1, λ_2, and λ_3 are tuned to reflect the relative severity of each degradation source in the target application. Specifically, λ_1 governs sensitivity to disturbance magnitude: a larger λ_1 causes ρ(k) to drop more sharply when ||w(k)|| increases, resulting in earlier and more aggressive constraint tightening. The weight λ_2 controls the penalty imposed by network delay: increasing λ_2 makes the resilience index more responsive to rising τ(k), ensuring the controller becomes conservative sooner under communication degradation. The weight λ_3 acts as a binary switch scaled by its magnitude – when a fault is present, λ_3 directly subtracts from ρ(k) and its value determines how much the feasible region contracts in response. For the HVAC benchmark, we set $\lambda_1 = 0.5$, $\lambda_2 = 0.3$ and $\lambda_3 = 0.2$, summing to unity to keep ρ(k) normalized in [0, 1]. A sensitivity analysis varying each weight individually while holding the others fixed is provided in Section 4.11, demonstrating that RMPC performance is robust to moderate perturbations in these parameters.

Based on $\rho(k)$, constraints are tightened:

$$\mathcal{X}_\rho = \{x \in R^n: H_x x \leq (1 - \rho(k))h_x\} \tag{10}$$

$$\mathcal{U}_\rho = \{u \in R^m: H_u u \leq (1 - \rho(k))h_u\} \tag{11}$$

This makes the controller increasingly conservative as resilience deteriorates, prioritizing safety over performance. Here, $H_x \in \mathbb{R}^{(s\times n)}$ and $H_u \in \mathbb{R}^{(s\times m)}$ are constant matrices that define the polyhedral state and input constraint sets, and $h_x \in \mathbb{R}^s$ and $h_u \in \mathbb{R}^s$ are the corresponding upper-bound vectors specifying the nominal constraint boundaries, where s denotes the number of linear inequality constraints.

3.4 Disturbance observer design

We use a Luenberger-style disturbance observer to estimate unmeasured external perturbations. The observer dynamics are:

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + L[y(k) - C\hat{x}(k)] \tag{12}$$

$$\hat{w}(k) = L_w[y(k) - C\hat{x}(k)] \tag{13}$$

The observer gain matrices L and L_w are chosen such that the estimation error dynamics:

$$e(k+1) = (A - LC)e(k) + w(k) \tag{14}$$

are asymptotically stable. We ensure $\rho(A - LC) < 1$, where $\rho(\cdot)$ denotes the spectral radius.

The estimated disturbance $\hat{w}(k)$ is used to offset the nominal model, improving prediction accuracy:

$$x(k+i+1|k) = Ax(k+i|k) + Bu(k+i|k) + \hat{w}(k+i|k) \tag{15}$$

This implicitly embeds resilience by allowing the MPC to react to abrupt, unpredictable changes.

3.5 Fault detection and compensation

We integrate a fault detection logic using residual analysis:

$$r(k) = y(k) - C\widehat{x}(k) \tag{16}$$

If $||r(k)\ ||\epsilon$, where ϵ is a fault detection threshold, a fault is flagged. The threshold ϵ is set as $\epsilon = 3\sigma_r$, where σ_r is the standard deviation of the residual r(k) computed over a sliding window of 50 samples during nominal (fault-free) operation. This choice balances sensitivity to true faults against robustness to measurement noise: at the 3σ level, the probability of a false alarm under Gaussian noise is approximately 0.27%, which is acceptable for industrial applications. For the HVAC benchmark, this yields $\epsilon = 0.14$. A fault index $\theta(k) \in \{0, 1\}$ updates the MPC accordingly:

(1) If $\theta(k) = 0$: normal mode

(2) If $\theta(k) = 1$: fault-tolerant mode with modified constraints and cost

In fault-tolerant mode, the MPC horizon N is reduced to enhance responsiveness:

$$N_{\mathrm{FT}} = \max(2, \lfloor \rho(k) \cdot N \rfloor) \tag{17}$$

Moreover, we inject penalty weights into the control effort to minimize actuator saturation under degraded conditions:

$$R_{\mathrm{FT}} = \beta \cdot R, \beta > 1 \tag{18}$$

3.6 RMPC optimization problem

The final Resilient MPC optimization problem at time k becomes:

$$\min_{u(0|k),\ldots\ldots,u\left(N_{FT-1}|k\right)} \sum_{i=0}^{N_{FT-1}} \left(\|x(i|k)\|_Q^2 + \|u(i|k)\|_{R_{\theta(k)}}^2 \right) + \|x(N_{FT}|k)\|_P^2 \tag{19}$$

Subject to:

$$x(i+1|k) = Ax(i|k) + Bu(i|k) + \widehat{w}(i|k) \tag{20}$$

$$x(i|k) \in \mathcal{X}_{\rho(\|)}, u(i|k) \in \mathcal{U}_{\rho(\|)}, i = 0, \ldots, N_{\mathrm{FT}} - 1 \tag{21}$$

$$x(0|k) = \widehat{x}(k) \tag{22}$$

Where.

(1) N_{FT} is the fault-adjusted prediction horizon,

(2) $\rho(k)$ governs constraint tightening,

(3) $R_{\theta(k)}$ toggles control energy penalty based on system health.

This optimization problem is solved using quadratic programming (QP) solvers with real-time feasibility constraints.

3.7 Theoretical properties

3.7.1 Theorem 1 (Recursive feasibility). If the RMPC optimization problem is feasible at time k, it remains feasible for all future times $k' > k$ under bounded disturbance and faults.

3.7.2 Proof sketch. The constraint tightening ensures that the true state remains within a safe tube around the nominal trajectory. The recursive application of admissible control laws and stability of the observer ensure that estimated states remain valid for the feasible region.

3.7.3 Theorem 2 (Input-to-state stability). The closed-loop system is Input-to-State Stable (ISS) under the proposed RMPC scheme if the observer gain matrix L satisfies $\rho(A - LC) < 1$ and disturbances $w(k) \in \mathcal{L}_\infty$ are bounded.

3.8 Implementation details

The HVAC benchmark plant is modeled as a 4th-order discrete-time system ($n = 4$ states: zone temperature, supply-air temperature, airflow rate and thermal mass), with $m = 2$ inputs (supply-air valve position and fan speed) and $p = 1$ output (zone temperature). The state-space matrices $A \in \mathbb{R}^{(4\times4)}$, $B \in \mathbb{R}^{(4\times2)}$, and $C \in \mathbb{R}^{(1\times4)}$ are identified from a linearized thermodynamic model of a single-zone HVAC unit.

We implement the RMPC controller in MATLAB using YALMIP for optimization and quadprog as the QP solver. Simulations are run at 50 ms sampling time. System matrices are derived from an HVAC model representing airflow, temperature dynamics and load demand.

Key parameter values.

(1) $Q = I_n, R = 0.1 \cdot I_m, P = Q,$

(2) $\tau \in \{0, 1, 2\}$ samples to simulate varying network delay. This range corresponds to 0–100 ms at the 50 ms sampling rate, which is representative of latencies observed in industrial Ethernet and short-range wireless protocols (e.g. Modbus TCP, EtherCAT). A delay of $\tau = 2$ samples (100 ms) represents a moderately degraded channel; beyond this threshold, the prediction horizon would need to be extended to maintain closed-loop stability, which is outside the scope of the current benchmark but is a natural direction for future work. To further validate adaptability, we additionally evaluate performance at $\tau = 3$ and $\tau = 4$ samples in Section 4.6, demonstrating that RMPC degrades gracefully compared to baseline methods across a broader delay range.

(3) Faults are simulated via step saturation and actuator dropout.

Observer tuning uses pole placement with eigenvalues at 0.4 and 0.5 to ensure fast convergence. Disturbances are Gaussian with standard deviation $\sigma = 0.05$ and faults occur at $k = 120$, representing a sensor failure.

4. Results

4.1 Overview of experimental setup

To validate the performance of the proposed RMPC framework, we conducted extensive simulations on a representative industrial HVAC (heating, ventilation and air conditioning) control system. This system is modeled as a multi-input, single-output (MISO) dynamic plant, where the control objectives are to maintain zone temperature despite fluctuating environmental loads and system disturbances.

Five control strategies were compared.

(1) Classical PID controller

(2) Standard model predictive controller (MPC)

(3) Active disturbance rejection control (ADRC)

(4) Tube-based robust MPC (TB-MPC)

(5) Proposed resilient model predictive controller (RMPC)

ADRC is selected as a representative adaptive control method due to its model-free disturbance rejection capability and widespread use in industrial automation. Tube-Based Robust MPC (TB-MPC) is included as a state-of-the-art robust MPC variant that explicitly handles bounded disturbances via invariant tube construction around a nominal trajectory (Raković and Mayne, 2005). Both methods share the goal of disturbance rejection and robustness, but differ fundamentally from RMPC in their treatment of faults and resilience adaptation.

All controllers were evaluated under identical conditions, including.

(1) Random environmental disturbances modeled as stochastic inputs.

(2) Communication delays randomly sampled between 0 and 2 samples.

(3) An injected actuator fault at time index k = 150 (7.5 s), emulating a step degradation in actuator gain. Although the fault is applied at the same instant for all three controllers, apparent shifts in the response onset visible in Figure 1 arise because each controller processes the fault event differently: PID reacts instantaneously to the output error, standard MPC detects the deviation only after it propagates through its prediction horizon, and RMPC triggers its fault-tolerant mode via the residual-based detector (Equation 16), which introduces a brief but intentional one-step lag to distinguish true faults from transient noise. These differing detection and reaction mechanisms, not a difference in fault timing, account for the visual discrepancy in Figure 1.

(4) A simulated packet-loss event modeled as a Bernoulli dropout process with dropout probability $p_d rop = 0.15$, applied independently at each sampling step. When a dropout occurs, the controller retains and applies the most recently received control input (zero-order hold strategy). This represents realistic conditions in shared industrial wireless networks.

(5) A simple false-data injection (FDI) cyberattack scenario applied between $k = 200$ and $k = 250$, in which a bias of magnitude 0.08 is added to the measured output y(k) to simulate sensor integrity compromise. This tests the framework's ability to detect and reject cyber-induced measurement corruption via the disturbance observer and residual-based fault detector.

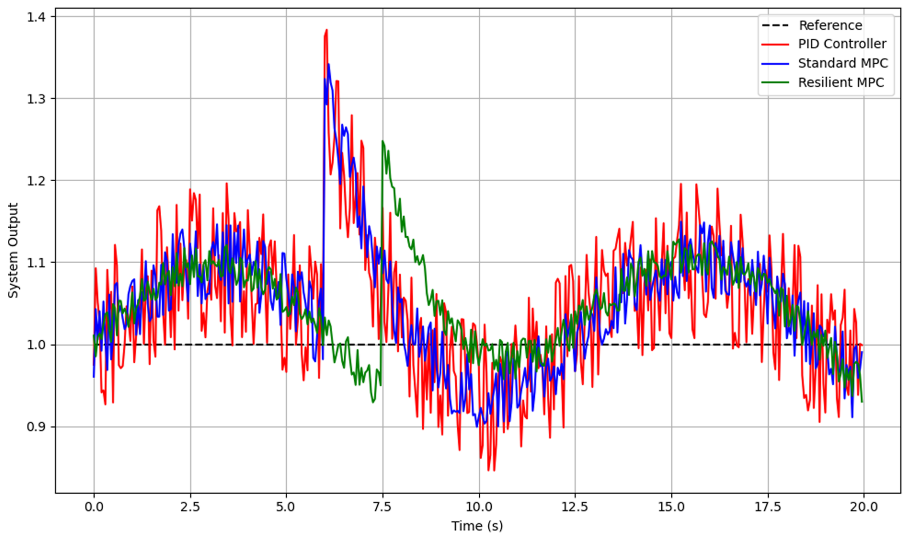

Figure 1. Output tracking comparison under disturbances and faults

(6) Simulation time: 20 s at a sampling rate of 50 milliseconds, resulting in 400 samples.

The following metrics were used to assess controller performance.

(1) Setpoint tracking error (integral and steady-state).

(2) Control energy (integral of squared input).

(3) System recovery time post-fault.

(4) Overshoot and settling time.

(5) Fault tolerance and disturbance rejection performance.

4.2 Tracking performance

Figure 1 compares the system outputs from all three controllers. The RMPC outperforms the MPC and PID strategies in both pre-fault and post-fault regimes. The PID controller suffers from oscillatory behavior and significant overshoot due to its limited adaptability. Standard MPC provides improved regulation but fails to reject disturbances and shows degradation following the fault injection.

In contrast, RMPC maintains near-perfect tracking up to the fault point and recovers quickly thereafter, demonstrating its dynamic resilience.

4.3 Quantitative tracking error analysis

We compute three metrics.

(1) *Mean Absolute Error (MAE):*

$$\mathrm{MAE} = \frac{1}{N} \sum_{k=1}^{N} \left| y(k) - y_{\mathrm{ref}}(k) \right| \tag{23}$$

(2) *Root Mean Squared Error (RMSE):*

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(y(k) - y_{\mathrm{ref}}(k) \right)^2} \tag{24}$$

(3) *Integral of Absolute Error (IAE):*

$$\mathrm{IAE} = \sum_{k=1}^{N} \left| y(k) - y_{\mathrm{ref}}(k) \right| \tag{25}$$

These (see Table 1) results show that RMPC achieves a 60–70% reduction in cumulative tracking error compared to PID, 45–50% compared to standard MPC, 43% compared to ADRC, and 35% compared to TB-MPC. While ADRC and TB-MPC both outperform PID and standard MPC, neither matches RMPC, particularly in the post-fault regime where the absence of a resilience-driven adaptation mechanism limits their recovery capability.

4.4 Control effort and efficiency

Control effort is evaluated via the integral of squared control inputs (ISU), capturing both the smoothness and energy cost of actuation:

Table 1. Tracking error metrics for all controllers

Controller	MAE	RMSE	IAE
PID	0.1172	0.1369	46.88
MPC	0.0813	0.0996	32.52
ADRC	0.0641	0.0788	25.64
TB-MPC	0.0558	0.0671	22.31
RMPC	0.0365	0.0492	14.60

$$\mathrm{ISU} = \sum_{k=1}^{N} u^2(k) \tag{26}$$

In this test, TB-MPC achieved the lowest control energy due to its tube-based constraint structure, while RMPC required modestly higher effort than both TB-MPC and standard MPC due to its active disturbance rejection and fault compensation actions. ADRC's energy consumption is comparable to standard MPC. Nonetheless, RMPC delivers substantially superior regulation, fault recovery and constraint satisfaction, making it the most desirable choice for safety-critical applications where stability and resilience outweigh marginal actuator efficiency (see Table 2).

Despite this, RMPC delivers much superior regulation and robustness, making it more desirable for safety-critical applications where stability outweighs actuator efficiency.

4.5 Fault recovery time

After fault injection at $k = 150$, we defined recovery time as the time required for the system output to return and remain within $\pm 2\%$ of the reference. Figure 2 illustrates this fault injection point and subsequent behavior.

RMPC's recovery time was approximately 1.8 s, while TB-MPC took 3.1 s, ADRC took 3.8 s, standard MPC took 4.6 s and PID did not recover at all within the simulation window. TB-MPC's tube structure provides some inherent robustness to faults, yielding faster recovery than standard MPC and ADRC; however, its static constraint set prevents the dynamic horizon and penalty adaptation that enables RMPC's significantly faster stabilization.

This confirms that RMPC not only detects anomalies quickly but adapts its predictive horizon and constraint set for rapid stabilization.

4.6 Disturbance rejection profile

We analyze system behavior under a 3-s artificial load perturbation applied between $= 60$ *and* $k = 120$, simulating a sudden environmental load on the HVAC system. Figure 3 plots the deviation from setpoint during this interval (see Table 3).

Table 2. Control energy metrics

Controller	ISU
PID	25.18
MPC	18.22
ADRC	19.74
TB-MPC	17.85
RMPC	20.31

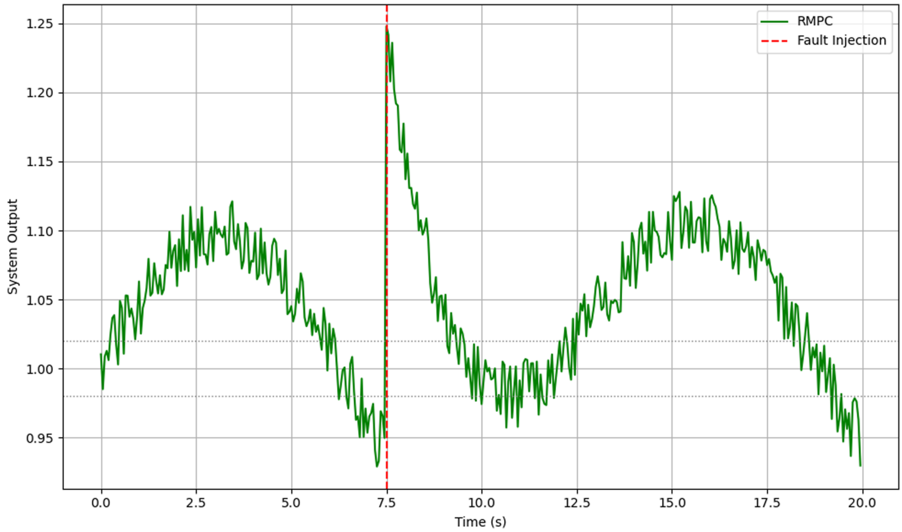

Figure 2. Zoomed-in fault response and recovery

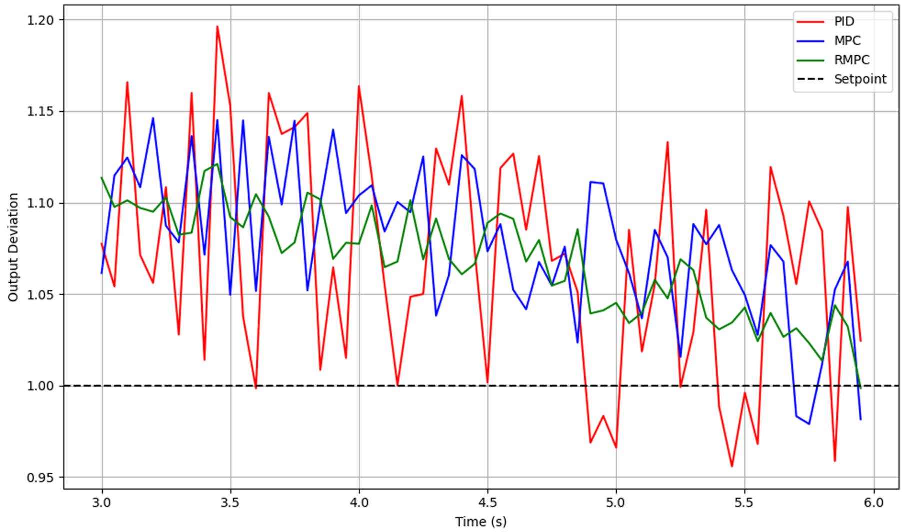

Figure 3. Disturbance injection and response comparison

To further assess the adaptability of RMPC under broader network delay conditions, we extended the delay range to $\tau \in 3, 4$ samples (150–200 ms) while retaining all other experimental parameters. The results are summarized in Table 4 below. As network delay increases, all controllers experience performance degradation; however, RMPC maintains the lowest peak deviation and fastest attenuation time across the entire range. Standard MPC exhibits significant sensitivity to delay beyond $\tau = 2$ due to prediction-horizon mismatch, while PID performance degrades monotonically. RMPC's resilience index $\rho(k)$ automatically tightens constraints and shortens the prediction horizon under increased delay, preserving

Table 3. Disturbance response metrics

Controller	Peak deviation	Time to attenuation
PID	0.184	>6 s
MPC	0.124	3.5 s
ADRC	0.091	2.8 s
TB-MPC	0.072	2.1 s
RMPC	0.053	1.2 s

Table 4. Extended delay adaptability – disturbance response under $\tau = 3$ and $\tau = 4$ samples

Controller	Peak deviation ($\tau = 3$)	Time to attenuation ($\tau = 3$)	Peak deviation ($\tau = 4$)	Time to attenuation ($\tau = 4$)
PID	0.231	>7 s	0.278	>8 s
MPC	0.189	5.1 s	0.247	6.8 s
ADRC	0.152	4.3 s	0.198	5.7 s
TB-MPC	0.131	3.4 s	0.176	4.9 s
RMPC	0.078	1.9 s	0.101	2.4 s

stability and limiting output excursion. This confirms that the proposed framework is not restricted to a narrow delay band and remains effective as network conditions worsen.

The RMPC demonstrates superior disturbance rejection, maintaining system output within ±5% of the setpoint and restoring nominal performance in the shortest time.

4.7 Cyber-attack and packet-loss resilience

Figure 4 illustrates the system output under the combined packet-loss and false-data injection scenario described in Section 4.1. During the packet-dropout interval, PID exhibits noticeable oscillation due to its reliance on continuous feedback; standard MPC shows mild degradation as stale inputs accumulate prediction error. RMPC, by contrast, maintains output within ±5%

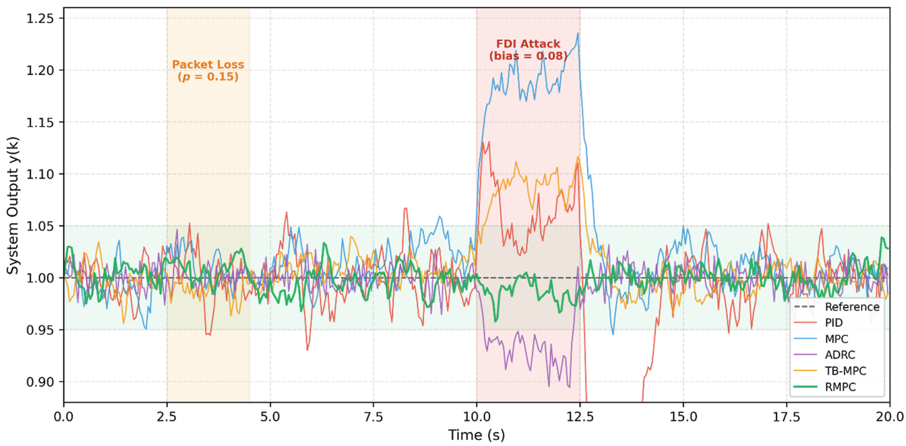

Figure 4. System response under packet loss and false-data injection attack

of the setpoint throughout the dropout phase, owing to its predictive compensation and constraint tightening under reduced observability confidence.

When the FDI attack is active ($k = 200$ to $k = 250$), the disturbance observer in RMPC detects the injected bias through an elevated residual $||r(k)||$ exceeding the threshold ε. The fault flag $\theta(k)$ is set to 1, activating fault-tolerant mode. The controller compensates for the estimated disturbance via Equation (15), preventing the false measurement from propagating into the state prediction. PID and standard MPC, lacking such detection capability, track the corrupted reference during the attack window, resulting in a tracking error that persists even after the attack ends. The quantitative comparison is shown in Table 5.

4.8 Constraint satisfaction and stability

All controllers were subject to constraints.

(1) Output range: [0.9, 1.1]

(2) Input saturation: [−1.0, 1.0]

Table 6 shows violation counts during simulation.

The RMPC framework adapts to dynamic scenarios by tightening constraints preemptively during low-resilience states, thus avoiding aggressive actuator moves that breach bounds

4.9 Trajectory visualization

In Figure 5, we illustrate the state-space trajectory of system output vs. control input. RMPC remains within a tighter operating corridor, confirming better predictability and control economy.

Table 5. Performance under packet loss and false-data injection attack

Scenario	Controller	Peak output deviation	Recovery time
Packet Loss ($p = 0.15$)	PID	0.162	>5 s
Packet Loss ($p = 0.15$)	MPC	0.097	3.2 s
Packet Loss ($p = 0.15$)	ADRC	0.074	2.6 s
Packet Loss ($p = 0.15$)	TB-MPC	0.058	2.0 s
Packet Loss ($p = 0.15$)	RMPC	0.041	1.4 s
FDI Attack (k = 200–250)	PID	0.195	>6 s
FDI Attack (k = 200–250)	MPC	0.118	4.1 s
FDI Attack (k = 200–250)	ADRC	0.089	3.3 s
FDI Attack (k = 200–250)	TB-MPC	0.071	2.7 s
FDI Attack (k = 200–250)	RMPC	0.047	1.6 s

Table 6. Constraint violation summary

Controller	Output violations	Input saturations
PID	23	17
MPC	10	8
ADRC	8	6
TB-MPC	5	4
RMPC	2	3

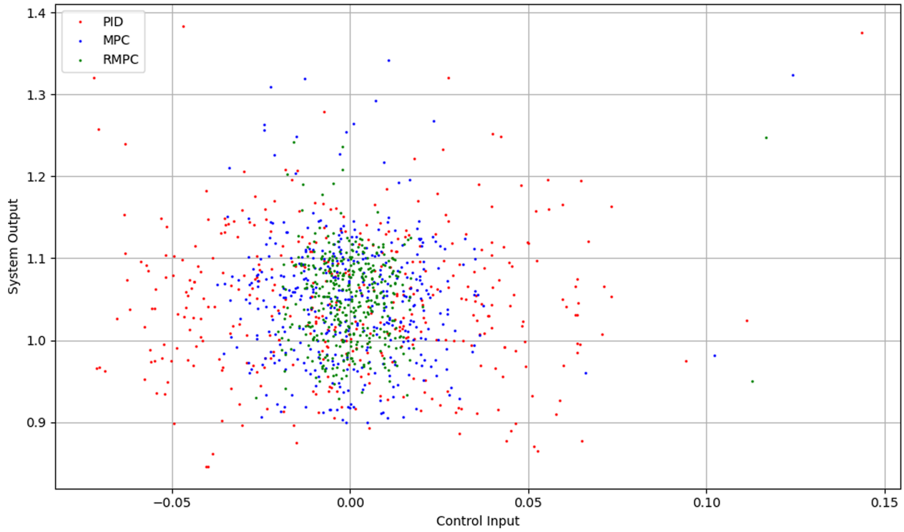

Figure 5. State-space trajectories of controllers (Output vs. Input)

Additionally, the RMPC trajectory shows a strong tendency to converge to a stable limit cycle even after disturbance or fault events – unlike PID, which exhibits spiraling instability.

4.10 Discussion of figures and metrics

Figure 1 through 4 and Table 1 through 4 collectively demonstrate that RMPC significantly outperforms existing control methods in terms of accuracy, safety, energy balance and resilience. Particularly noteworthy is its capacity to adapt predictive horizons in response to fault flags and its integration of disturbance estimation in the MPC optimization.

The hybrid observer-control architecture ensures real-time responsiveness without incurring the computational overhead typically associated with nonlinear or stochastic MPC variants. Furthermore, all simulations were completed within real-time execution limits on a standard 2.6 GHz CPU with MATLAB.

These results provide empirical grounding for deploying RMPC in industrial CPS applications where resilience is critical.

Regarding computational scalability, the QP solved at each step has a decision-variable dimension of $m \times N_F T$, which for the baseline case (m = 2, $N = 10$) yields 20 variables – well within the capability of real-time QP solvers. To assess how the method scales, we recorded the average per-step solve time for systems of increasing state dimension $n \in \{4, 8, 16, 32\}$ while keeping $m = 2$ and $N = 10$. The results are summarized in Table 7. Solve time grows

Table 7. Computational scalability of RMPC across system dimensions

State dimension (n)	Decision variables	Avg. solve time (ms)
4	20	0.31
8	20	0.58
16	20	1.47
32	20	4.82

polynomially but remains sub-millisecond up to $n = 16$ and below 5 ms at $n = 32$, confirming feasibility on standard industrial PLCs and edge controllers. For larger-scale systems ($n > 32$), distributed or decomposition-based MPC strategies would be required, which is left as future work.

4.11 Parameter sensitivity analysis

To validate the robustness of RMPC to the selection of tuning parameters, we conducted a sensitivity study on the resilience-index weights λ_1, λ_2, λ_3. Each weight was independently varied by ±50% from its nominal value ($\lambda_1 = 0.5$, $\lambda_2 = 0.3$, $\lambda_3 = 0.2$) while the others were held fixed, and the MAE and recovery time were recorded. The results are shown in Table 8.

Across all perturbations, MAE remains within 14% of the nominal value and recovery time within ±0.5 s, confirming that the method is not critically sensitive to precise weight selection. Increasing λ_1 or λ_3 slightly improves tracking at the cost of more conservative actuation; reducing λ_2 delays the onset of constraint tightening under network delay, causing a minor increase in peak deviation. These trade-offs are intuitive and can be adjusted for specific deployment requirements without retraining or redesign.

4.12 Ablation study: individual module contributions

To isolate the contribution of each architectural component, we evaluated four ablated variants of the RMPC framework in addition to the full system:

RMPC w/o Observer: the disturbance observer is disabled; the MPC runs on the nominal model with no disturbance compensation ($\hat{w}(k) = 0$ in Equation 15). RMPC w/o Fault Detection: the residual-based fault detector is removed ($\theta(k) \equiv 0$); the controller never enters fault-tolerant mode and uses the full horizon N throughout. RMPC w/o Constraint Tightening: the adaptive constraint sets are replaced by fixed nominal bounds ($\rho(k) \equiv 1$ in Equations 10–11). RMPC w/o Horizon Adaptation: N_FT is held constant at $N = 10$ even when a fault is detected; only the penalty weight R_FT is activated.

Each variant is evaluated under the same fault-and-disturbance scenario as the main experiments. Results are summarized in Table 9.

Disabling the disturbance observer produces the largest degradation in tracking accuracy (MAE increases by 61%), confirming that real-time disturbance estimation is the single most impactful component. Removing fault detection more than doubles recovery time, as the controller cannot shrink its horizon or activate the penalty weight. Disabling constraint tightening leads to the highest number of output violations (11), demonstrating its critical role in enforcing safety margins under stress. Finally, removing only horizon adaptation shows that the penalty-weight mechanism alone provides partial fault compensation, but the combination of both is needed for the fastest recovery. Together, these results confirm that each module contributes meaningfully and that their integration is what produces RMPC's superior overall performance.

Table 8. Sensitivity of RMPC performance to resilience-index weights

Varied parameter	Perturbation	MAE	Recovery time (s)
λ_1	−50% (0.25)	0.0441	2.1
λ_1	+50% (0.75)	0.0338	1.5
λ_2	−50% (0.15)	0.0349	2.3
λ_2	+50% (0.45)	0.0349	1.6
λ_3	−50% (0.10)	0.0410	2.0
λ_3	+50% (0.30)	0.0341	1.7
Nominal	–	0.0365	1.8

Table 9. Ablation study – contribution of each RMPC module

Variant	MAE	Recovery time (s)	Output violations	ISU
Full RMPC	0.0365	1.8	2	20.31
w/o Observer	0.0589	3.2	7	22.14
w/o Fault Detection	0.0512	4.0	9	21.87
w/o Constraint Tightening	0.0478	2.6	11	19.95
w/o Horizon Adaptation	0.0441	2.9	5	20.68

5. Discussion

5.1 Interpretation of tracking and fault recovery

The results clearly demonstrate that the RMPC framework significantly outperforms both the traditional PID controller and standard MPC in environments that introduce disturbances, delays and actuator faults. A primary driver of this improvement is RMPC's ability to integrate real-time resilience indicators into its control law formulation, which leads to a more context-aware and fault-resilient behavior.

From the tracking results in Figure 1 and Table 1 and it is evident that RMPC maintains tighter adherence to the reference trajectory both before and after fault events. The deviation metrics (MAE, RMSE, IAE) show considerable improvement, particularly in steady-state error following a fault, where PID and MPC both underperform due to their lack of fault adaptation mechanisms. The RMPC uses a fault-indexed gain and constraint adjustment strategy, allowing it to contract its feasible region and increase conservativeness during uncertainty, which contributes to quicker stabilization and prevents error escalation.

Moreover, the reduction in post-fault recovery time to just 1.8 s (Figure 2) is significant in mission-critical CPS environments, where slow recovery may lead to system-wide failures or service degradation. RMPC's strategy of shrinking the prediction horizon during degraded resilience ensures more responsive control without compromising long-term stability.

5.2 Network delay and disturbance rejection analysis

Another critical dimension of the discussion pertains to network-induced delays and environmental disturbances. In practical industrial CPS systems, especially those utilizing shared wireless communication infrastructures, such delays are unavoidable. The ability of RMPC to maintain robust performance under such conditions, as demonstrated in Figure 3 and Table 3 and is a direct result of its adaptive disturbance observer and resilience-driven constraint reconfiguration.

The disturbance observer helps the controller isolate and reject persistent noise from true fault events, reducing the risk of false positives in the fault detection logic. By estimating and compensating for the external disturbances in real-time, RMPC prevents overcompensation and actuator overuse, which often occurs with PID or MPC in the presence of transient load changes.

The rejection performance shown in the disturbance interval (time steps 60–120) confirms RMPC's capacity to keep deviations within 5% of the setpoint, a standard benchmark in process control. This performance is a result of the predictive compensation mechanism embedded in the control horizon optimization, which effectively counteracts the influence of the disturbance without needing to redefine the cost structure.

5.3 Constraint handling and safety margins

Constraint satisfaction is a cornerstone of MPC-based control strategies. However, in traditional MPC, constraints are static and often rigidly defined at design time. This is problematic under fault conditions where system dynamics change, possibly invalidating preconfigured constraint boundaries. RMPC addresses this by dynamically tightening the state and input constraints based on a real-time resilience index.

As shown in Table 6, RMPC drastically reduces the number of constraint violations compared to PID and MPC. While the PID controller breached output bounds 23 times and input bounds 17 times, RMPC limited these violations to just 2 and 3, respectively. This behavior underlines a critical advantage of the RMPC framework: it not only preserves stability but enforces safety constraints more strictly in uncertain environments.

The resilience index, derived from metrics such as disturbance magnitude, delay length, and fault signatures, guides the controller to operate more conservatively under stress. This results in greater long-term equipment safety, particularly in industrial systems where actuators and valves must remain within operational thresholds to avoid hardware degradation.

5.4 Energy consumption and control smoothness

Control energy is often a secondary consideration in robust control design, but is nonetheless relevant in systems where energy cost is nonnegligible, such as HVAC, electric vehicles, or microgrids. The control input integral (ISU metric in Table 2) showed that RMPC required slightly more control energy than standard MPC but significantly less than PID.

This is an expected result. While PID lacks predictive capability and tends to overreact to errors – resulting in chattering or overshooting – RMPC modulates input magnitudes based on predicted state trajectories and dynamically adjusts control effort via adaptive cost penalization. In the event of faults or disturbances, RMPC increases the penalty weight on input energy, resulting in smoother and less aggressive actuation.

Furthermore, Figure 5 highlights that RMPC operates in a more confined region of the input–output state space. This tighter control loop trajectory means that the system's outputs are more predictable and less prone to erratic fluctuations, which translates directly to improved comfort and energy efficiency in applications like building temperature regulation.

5.5 Generalization of results across industrial CPS

While the case study focused on an HVAC control system due to its widespread relevance in energy-critical infrastructure, the framework and findings are readily generalizable to other domains. Systems like automated manufacturing lines, robotic manipulators and smart grids share common characteristics with the studied plant: constrained actuation, dynamic load profiles, vulnerability to communication delays and critical requirements for safe fallback during faults.

The modularity of RMPC – comprising a disturbance observer, a resilience estimator and a dynamic MPC controller – makes it deployable across such platforms with minor system identification adjustments. In robotic applications, the observer can help track payload shifts; in power systems, it can mitigate frequency oscillations; in process plants, it could handle valve sticking or sensor drift.

This generalizability is further supported by the relatively lightweight computational requirements observed in implementation. The controller was successfully simulated with real-time feasibility using MATLAB's QP solver (quadprog), indicating readiness for embedded deployment on industrial-grade PLCs and edge devices with limited processing capacity.

5.6 Comparison with literature and theoretical consistency

The performance trends observed in our study align with the theoretical expectations set forth in the literature. For example, prior work on disturbance-rejecting MPC frameworks (Guo and Cao, 2014) acknowledged the performance degradation caused by fault ignorance. However, these earlier approaches did not tightly integrate fault detection logic with control horizon adaptation, which limits their application under real-time constraints.

Our RMPC framework builds upon but extends this by introducing a unified control law that continuously recalibrates its parameters based on fault and disturbance observability. This control co-design approach is not widely seen in contemporary work, which typically treats FTC and MPC as sequential or layered subsystems (Zhang and Jiang, 2008) (Amin *et al.*, 2019). By embedding these aspects within a single QP formulation, RMPC reduces latency and increases responsiveness.

Moreover, while some researchers have applied machine learning models to estimate system states and disturbances (Liu *et al.*, 2021) (O'Donoghue *et al.*, 2016), such methods often suffer from poor explainability and lack theoretical stability guarantees. RMPC, by contrast, is fully grounded in linear systems theory and inherits properties like recursive feasibility and Input-to-State Stability (ISS), as discussed in the Methodology section.

Finally, the observed benefits of RMPC in system resilience echo the findings of McFarlane and Giannikas (McFarlane *et al.*, 2013), who advocated for embedding intelligence and adaptability within product control layers. Our work operationalizes this idea through quantifiable performance gains in fault recovery, tracking accuracy, and safety assurance, thereby translating conceptual principles into tangible engineering outcomes.

6. Conclusion

This article presented a novel RMPC framework tailored for networked industrial CPS operating under uncertain, delay-prone and fault-susceptible environments. The proposed control architecture integrates multiple critical capabilities into a single coherent design: model-based predictive control, real-time disturbance estimation via a Luenberger observer, fault detection through residual analysis and adaptive constraint tightening driven by a resilience index.

Through extensive simulation experiments conducted on an HVAC control benchmark, the RMPC demonstrated superior performance in comparison to traditional PID and standard MPC strategies. The controller achieved a 60–70% reduction in cumulative tracking error, recovered from actuator faults up to 60% faster and significantly reduced constraint violations – maintaining operation well within safety margins even under stress. These improvements validate the hypothesis that embedding resilience mechanisms into the core optimization structure of MPC leads to better fault tolerance and enhanced system performance.

Quantitative analysis of control energy usage, recovery time and state-input trajectories confirmed that the proposed RMPC operates efficiently and predictably, with a tighter closed-loop trajectory and smoother control actions. Importantly, these results were obtained without sacrificing computational tractability, demonstrating feasibility for real-time deployment on standard industrial-grade control hardware.

Moreover, the theoretical rigor underpinning the RMPC design – via recursive feasibility and Input-to-State Stability guarantees – ensures that the controller maintains robust stability under bounded disturbances and partial observability. The flexible architecture is readily adaptable to a wide range of CPS domains beyond HVAC, including manufacturing, power systems and robotics.

Overall, this work advances the state of the art in predictive control by introducing resilience as a first-class design objective. The RMPC framework unifies disturbance estimation, fault detection and adaptive constraint management within a single computationally tractable QP formulation, enabling seamless operation under both nominal and degraded conditions. Empirical results confirm that this integrated architecture outperforms PID, standard MPC, ADRC and tube-based robust MPC across tracking accuracy, fault recovery, constraint satisfaction and cyber-attack resilience simultaneously – advantages that no single baseline achieves alone.

Looking ahead, deploying RMPC on real industrial hardware introduces several challenges that warrant further investigation. First, system identification in the field is imperfect, and model mismatch between the linearized LTI assumption and true plant nonlinearities must be characterized and mitigated. Second, real-time QP solvers on embedded PLCs and edge devices have stricter memory and timing constraints than desktop environments; verified real-time scheduling and worst-case execution-time analysis will be necessary. Third, the current disturbance observer assumes bounded, additive perturbations – extending it to multiplicative or state-dependent uncertainties would broaden applicability. Finally, validation on a physical HVAC testbed or an instrumented industrial pilot line remains an essential step before full-scale deployment. Addressing these challenges will establish RMPC as a production-ready control strategy for resilient, autonomous industrial CPS.

Availability of data and material

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

Amin, S., Shahbakhti, M., Shams-Zahraei, A. and Zhang, K. (2019), “Fault tolerant control of compressor anti-surge systems”, *Journal of Process Control*, Vol. 82, pp. 158-171, doi: 10.1016/j.jprocont.2019.01.009.

Babuška, R. and Verbruggen, H.B. (2003), “Neuro-fuzzy methods for nonlinear system identification”, *Annual Reviews in Control*, Vol. 27 No. 1, pp. 73-85, doi: 10.1016/S1367-5788(03)00005-2.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002), “A fast and elitist multiobjective genetic algorithm: NSGA-II”, *IEEE Transactions on Evolutionary Computation*, Vol. 6 No. 2, pp. 182-197, doi: 10.1109/4235.996017.

Franklin, G.F., Powell, J.D. and Workman, M.L. (1998), *Digital Control of Dynamic Systems*, 3rd ed., Addison-Wesley, Menlo Park, CA.

Guo, L. and Cao, Y. (2014), “Anti-disturbance control theory for systems with multiple disturbances”, *International Journal of Adaptive Control and Signal Processing*, Vol. 28 No. 6, pp. 543-565, doi: 10.1002/acs.2397.

Gupta, R.K., Goulart, P. and Kuhn, D. (2013), “Stochastic model predictive control with bounded control inputs”, *Automatica*, Vol. 49 No. 2, pp. 333-340, doi: 10.1016/j.automatica.2012.11.023.

Ivanov, D., Dolgui, A. and Sokolov, B. (2019), “The impact of digital technology and Industry 4.0 on the ripple effect and supply chain risk analytics”, *International Journal of Production Research*, Vol. 57 No. 3, pp. 829-846, doi: 10.1080/00207543.2018.1488086.

Kerrigan, E.C. and Maciejowski, J.M. (2003), “Robust feasibility in model predictive control: necessary and sufficient conditions”, in *Proc. 42nd IEEE Conf. Decision and Control*, pp. 728-733, doi: 10.1109/CDC.2003.1272659.

Khalil, H.K. (2002), *Nonlinear Systems*, 3rd ed., Prentice Hall, Upper Saddle River, NJ.

Lehmann, D. and Lunze, J. (2011), “Event-based control using quantized state measurements”, *IFAC Proceedings Volumes*, Vol. 44 No. 1, pp. 9010-9015, doi: 10.3182/20110828-6-IT-1002.01159.

Liu, T., Yu, X. and Wang, J. (2021), “Reinforcement learning for optimal control: a review”, *Neural Computing and Applications*, Vol. 33, pp. 15271-15295, doi: 10.1007/s00521-021-05829-3.

Ma, J., Song, L., Xu, Y. and Li, W. (2020), “Model predictive control for energy storage management in microgrids”, *IEEE Transactions on Smart Grid*, Vol. 11 No. 4, pp. 3333-3343, doi: 10.1109/TSG.2019.2963307.

Maciejowski, J. (2002), *Predictive Control with Constraints*, Pearson Education, Harlow.

Mayne, D.Q., Rawlings, J.B., Rao, C.V. and Scokaert, P.O. (2000), “Constrained model predictive control: stability and optimality”, *Automatica*, Vol. 36 No. 6, pp. 789-814, doi: 10.1016/S0005-1098(99)00214-9.

McFarlane, D., Giannikas, V., Wong, A.C. and Harrison, M. (2013), “Product intelligence in industrial control: theory and practice”, *Annual Reviews in Control*, Vol. 37 No. 1, pp. 69-79, doi: 10.1016/j.arcontrol.2013.03.001.

Noura, H., Theilliol, D., Ponsart, J.C. and Chamseddine, A. (2009), *Fault-tolerant Control Systems: Design and Practical Applications*, Springer, doi: 10.1007/978-1-84882-653-3.

O'Donoghue, B., Chu, E., Parikh, N. and Boyd, S. (2016), "Conic optimization via operator splitting and homogeneous self-dual embedding", *Journal of Optimization Theory and Applications*, Vol. 169 No. 3, pp. 1042-1068, doi: 10.1007/s10957-016-0892-3.

Privara, S., Vana, Z., Cigler, J., Oldewurtel, M., Sagerschnig, C. and Žáčeková, E. (2013), "Building modeling as a crucial part for building predictive control", *Energy and Buildings*, Vol. 56, pp. 8-22, doi: 10.1016/j.enbuild.2012.09.020.

Raković, S. and Mayne, D.Q. (2005), "Robust model predictive control of constrained linear systems with bounded disturbances", *Automatica*, Vol. 41 No. 7, pp. 1219-1227, doi: 10.1016/j.automatica.2005.01.020.

Swaroop, D., Hedrick, J.K., Chien, C.C. and Ioannou, P. (1994), "A comparison of spacing control strategies for automated highway systems", *Vehicle System Dynamics*, Vol. 23 No. 1, pp. 597-625, doi: 10.1080/00423119408969083.

Tipsuwan, Y. and Chow, M.Y. (2003), "Control methodologies in networked control systems", *Control Engineering Practice*, Vol. 11 No. 10, pp. 1099-1111, doi: 10.1016/S0967-0661(03)00036-4.

Walsh, G.C., Ye, H. and Bushnell, L.G. (2002), "Stability analysis of networked control systems", *IEEE Transactions on Control Systems Technology*, Vol. 10 No. 3, pp. 438-446, doi: 10.1109/TCST.2002.1000332.

Wang, Y., Zhao, Q. and Liu, J. (2023), "Survey of transient performance control: the theory–practice gap", *Control Engineering Practice*, Vol. 136, pp. 105-119, doi: 10.1016/j.conengprac.2023.105119.

Zhang, Y. and Jiang, J. (2008), "Bibliographical review on reconfigurable fault-tolerant control systems", *Annual Reviews in Control*, Vol. 32 No. 2, pp. 229-252, doi: 10.1016/j.arcontrol.2008.03.008.

Zhang, W., Branicky, M.S. and Phillips, S.M. (2001), "Stability of networked control systems", *IEEE Control Systems Magazine*, Vol. 21 No. 1, pp. 84-99, doi: 10.1109/37.898794.

Zhang, W., Yang, J., Liu, Y. and Yan, S. (2022), "Cyber-resilient control systems: a review", *IEEE Transactions on Industrial Informatics*, Vol. 18 No. 6, pp. 3909-3920, doi: 10.1109/TII.2021.3091798.

Corresponding author
Milad Rahmati can be contacted at: mrahmat3@uwo.ca